PyTradier Documentation

Release 0.0

Robert Leonard

May 31, 2023

Getting Started

getting_started

Please see the examples for a quick start with this library: Examples!.

CHAPTER 1

Risk Disclosure

Investing and trading through any medium contains a certain level of risk of loss. PyTradier is merely a tool for interfacing with the Tradier Brokerage. By using PyTradier, you agree to the terms and conditions of both Tradier's Brokerage and Development portal and the license associated with PyTradier. The creators and collaborators of PyTradier are not responsible for any fees, damages, losses, or other results associated with using or inability to use this software. Use PyTradier at your own risk.

It is your responsibility to carefully review the laws and regulations associated with this software and the activities associated with this software before attempting to use it.

See the applicable license for PyTradier on GitHub for further information.

CHAPTER 2

Installing PyTradier

As of right now, the only way to install PyTradier is by cloning the GitHub repository into your project.

In the future, PyTradier will be hosted by pypi.com for use with `pip install`. Until then, please clone the repository.

CHAPTER 3

Authenticating Tradier

Coming soon!

CHAPTER 4

Quick Start

PyTradier first needs to have the `Tradier` class initialized, which is how the program logs into the API. This is required since all of Tradier's API is private and requires authentication before accessing. This can be done using the following example:

```
import PyTradier

tradier = Tradier(token='abc123', account_id='123456', sandbox=False)
```

The initialization for `Tradier` takes three parameters:

- `token` Your API access token provided by Tradier. *Required.*
- `account_id` The ID number associated with your Tradier Brokerage account. You will only have this if you have opened a Brokerage account with Tradier. *Optional.*
- `endpoint` Determines whether the developer sandbox, brokerage sandbox, or full API endpoint will be used. Developer sandbox has limited access, but is completely free. The full API requires a Brokerage account. Brokerage accounts also come with a sandbox account for paper trading and has the capabilites of the full API. *Optional.*

Once the `Tradier` class has been initialized, all submodules can be called like this:

```
stocks = tradier.stock('AAPL', 'MSFT')
options = tradier.option('AAPL170630P00130000')
```

CHAPTER 5

Account

5.1 Account Balance

Coming soon!

CHAPTER 6

User

Coming soon!

CHAPTER 7

Market Information

The market directory allows programs to access information about the market, including searching for companies and retrieving market timings.

7.1 Market Calendar

7.1.1 Examples

```
# Retrieve a dictionary of the market open intervals
calendar = tradier.market.calendar()
print calendar.open()
```

The output is a dictionary with dates as keys and the status of the market on that particular date. Below is a shortened example output:

```
{u'2017-07-05': {u'start': u'09:30', u'end': u'16:00'}, u'2017-07-04': None, u'2017-
↪07-03': {u'start': u'09:30', u'end': u'13:00'}}
```

The `open()` function (as well as `premarket()` and `postmarket()`) returns a dictionary for each date in the main dictionary. This is useful since the information can be searched by its date (the key), or the entire dictionary can be looped through to get the values of each key. For example, to programmatically retrieve the market open interval from each day, use a `for` loop:

```
for key in calendar.open():
   print calendar.open()[key]  # loop through each nested dictionary
```

The (shortened) output of this is as follows:

```
None
{u'start': u'09:30', u'end': u'16:00'}
{u'start': u'09:30', u'end': u'16:00'}
```

7.2 Symbol Lookup

7.3 Company Search

7.4 Market Status

CHAPTER 8

Company Information

8.1 Options Chain

8.2 Historical Pricing

8.2.1 Examples

A year’s worth of historical prices can be retrieved using this example:

```
company = tradier.company(symbol='AAPL')
history = company.history(interval='daily', start='2011-1-1', end='2012-1-1')
print history.high()
```

8.3 Time and Sales

CHAPTER 9

Orders

Coming soon!

CHAPTER 10

Securities

The securities directory is an easy way to access market data for stocks and options. Both the `stock` and `option` classes inherit from the `quote` class.

10.1 Options

10.2 Stock

CHAPTER 11

Indices and tables

- genindex
- modindex
- search